

Aktuelle Konfiguration des MMDVM_HS_Dual_Hat als Duplex-Hotspot auf Basis der Pi-Star-Distribution

Firmwareanpassungen des Moduls MMDVM_HS_Dual_Hat als GPIO-Variante auf einem Raspberry Pi

Die Firmwareanpassungen sind *meine Empfehlungen*, wie man dann diese Firmware in das Modem einspielt habe ich [hier](#) umfassend erläutert. Deswegen beziehe ich mich hier nur auf die wesentlichen Definitionen in der **Config.h** und der **ADF7021.h** im Quellcode (https://github.com/juribeparada/MMDVM_HS). So wie ich das hier darstelle, habe ich das aktuell an meinen eigenen Hotspot(s) in Betrieb. Alles bezieht sich auf die aktuelle Version der Firmware.

Variante A mit aktivierter RX-AFC für 4-FSK (aktuell *nicht* empfohlen, AFC-Funktion ist problematisch):

Config.h:

```
#define MMDVM_HS_DUAL_HAT_REV10
#define ENABLE_ADF7021
#define DUPLEX
#define ADF7021_14_7456
#define AD7021_GAIN_AUTO_LIN
#define STM32_USART1_HOST
#define ENABLE_SCAN_MODE
#define SEND_RSSI_DATA
#define ENABLE_DEBUG
// Frequenzsperrern (falls gewünscht) ausschalten ab V1.4.12, bei < 1.4.12
// die nächsten zwei Optionen weglassen
#define DISABLE_FREQ_CHECK
#define DISABLE_FREQ_BAN
```

ADF7021.h:

```
#define ADF7021_DISABLE_RC_4FSK
#define ADF7021_ENABLE_4FSK_AFC
```

Variante B ohne RX-AFC für 4-FSK, aber abgeschalteten TX Raised Cosine Filter bei 4FSK Modulation wie DMR oder C4FM (**meine Empfehlung - lieferte bisher die besten Ergebnisse**):

Config.h:

```
#define MMDVM_HS_DUAL_HAT_REV10
#define ENABLE_ADF7021
#define DUPLEX
#define ADF7021_14_7456
#define AD7021_GAIN_HIGH
```

```
#define STM32_USART1_HOST
// Enable Nextion LCD serial port repeater on USART2 (ZUMspot Libre Kit and
ZUMspot RPi):
#define SERIAL_REPEATER
#define ENABLE_SCAN_MODE
#define SEND_RSSI_DATA
#define ENABLE_DEBUG
// Frequenzsperrern (falls gewünscht) ausschalten ab V1.4.12, bei < 1.4.12
die nächsten zwei Optionen weglassen
#define DISABLE_FREQ_CHECK
#define DISABLE_FREQ_BAND
```

ADF7021.h:

```
#define ADF7021_DISABLE_RC_4FSK
// #define ADF7021_ENABLE_4FSK_AFC
```

```
pi-star@dmr-hs-d11bz(rw):MMDVM_HS$ make clean
pi-star@dmr-hs-d11bz(rw):MMDVM_HS$ make
Upload gebaute Firmware ins Modem:
pi-star@dmr-hs-d11bz(rw):MMDVM_HS$ sudo make mmdvm_hs_dual_hat
Neustart:
pi-star@dmr-hs-d11bz(rw):MMDVM_HS$ sudo reboot
```

Kürzeres TXDelay beim Hotspot (Zuarbeit von Volker DL2SDG)

(getestet mit Firmware >= V1.4.12)

DMRDMOTX.cpp:

```
// m_txDelay(240U),          // 200ms
m_txDelay(120U),
```

```
void CDMRDMOTX::setTXDelay(uint8_t delay)
{
    // m_txDelay = 600U + uint16_t(delay) * 12U;          // 500ms + tx delay
    m_txDelay = 60U + uint16_t(delay) * 12U;
}
```

DMRTX.cpp:

```
// const uint32_t STARTUP_COUNT = 20U;
const uint32_t STARTUP_COUNT = 2U;
```

Das ist experimentell ermittelt (Zuarbeit Volker DL2SDG - danke) und liegt jetzt etwa auf dem Niveau eines Hytera-Repeater. Im Original gibts ca. 0,5s Leerlauf bis Daten kommen. Mit dieser Änderung dürfte sich das um 0,1s bewegen. Es könnte aber DMR-Geräte geben, die mit dieser Einstellung Probleme machen. Muss also jeder selber testen - das ist experimentell und zum Ausprobieren

gedacht.

Konfiguration MMDVMHost - dem Steuerprogramm für das MMDVM-Modem

Ich benutze als Plattform [Pi-Star](#), deswegen bezieht sich das Folgende auch auf **diese** Plattform. Wer das auf anderen Distributionen nutzt, muss das entsprechend anpassen. Ich hatte beim Abgleichen des Modems festgestellt, dass es eine Abweichung der TX/RX-Frequenz um +400Hz gab, der Parameter

```
TXOffset=-400
RXOffset=-400
```

korrigiert das entsprechend. Das muss aber an Eurem Hotspot ermittelt werden - die Werte können natürlich von meinen abweichen.

```
pi-star@dmr-hs-dl1bz(ro):~$ sudo -s
root@dmr-hs-dl1bz(ro):~# rpi-rw
root@dmr-hs-dl1bz(rw):~# nano /etc/mmdvmhost
```

/etc/mmdvmhost (nur die relevanten Sektionen):

```
[General]
Callsign=<dein_Rufzeichen_hier_eintragen>
Id=<deine_DMR_Id_hier_eintragen>
# Timeout entspricht Sprechzeitbegrenzung
Timeout=360
# wir wollen Duplex machen - wir haben ja 2 Timeslots
Duplex=1
RFModeHang=360
NetModeHang=360
Display=None
Daemon=1

[Info]
# Frequenzen definieren, Ablage -9,4 MHz, Bandplan und umliegende Relais
beachten !!! Bitte keine Repeater stören !
RXFrequency=430575000
TXFrequency=439975000
Power=0.01
Latitude=51.1711
Longitude=14.4193
Height=201
Location="Bautzen, J071FE"
Description="Germany"
URL=http://www.qrz.com/db/DL1BZ

[Modem]
Port=/dev/ttyAMA0
TXInvert=1
```

```
RXInvert=0
PTTInvert=0
TXDelay=100
RXOffset=-400
TXOffset=-400
DMRDelay=0
RXLevel=50
TXLevel=50
RXDCOffset=0
TXDCOffset=0
RFLevel=100
CWIdTXLevel=50
D-StarTXLevel=50
# DMRTXLevel=51 ermittelt nach Abgleich mit MMDVMCal
DMRTXLevel=51
# meine Erfahrung bisher: bei C4FM DMRTXLevel minus 2 oder 3 ist gleich
YSFTXLevel
YSFTXLevel=49
P25TXLevel=50
NXDNTXLevel=50
POCSAGTXLevel=50
RSSIMappingFile=/usr/local/etc/RSSI.dat
Trace=0
Debug=0

[DMR]
Enable=1
Beacons=0
BeaconInterval=60
BeaconDuration=3
ColorCode=1
SelfOnly=0
EmbeddedLCOnly=0
DumpTADData=1
# es gibt hier und da die Empfehlung CallHang=0 zu setzen - habe ich
getestet, danach hatte ich Probleme mit SMS-Versand und Empfang, mit
CallHang=3 klappte es wieder
# CallHang in sec bezeichnet die repeater-seitige GroupHangTime, die dem
DMR-Geraet uebermittelt wird und eigentlich ausgewertet werden sollte, macht
leider nicht jedes Geraet :(
CallHang=3
# TXHang ist die Sendernachlaufzeit in sec
TXHang=4
# ich verriegele den Mode DMR mit 2min Nachlauf
ModeHang=120

[DMR Network]
# diese Config ist für den Betrieb mit dem DMRGateway eingestellt !!!
Enable=1
Address=127.0.0.1
Port=62031
```

```
Jitter=360
Password="none"
# wir haben einen Duplex-Hotspot und können und wollen beide Timeslots
nutzen !!
Slot1=1
Slot2=1
Debug=0
ModeHang=120
Local=62032
```

root@dmr-hs-dl1bz(rw):~# reboot

Und nun testen und fleißig funken !

73 Heiko, DL1BZ

From:
<http://kb.amft-it.de/> - **Amateurfunk - Knowledge Base und Wiki by DL1BZ**

Permanent link:
<http://kb.amft-it.de/doku.php?id=kb-afu:mmdvm-hs-config>

Last update: **21.10.2019 11:25**

